

## Dồn bò [flowers]

Bò đi hái củi và để đàn bò gồm  $N$  con ăn cỏ trên cánh đồng như thường lệ. Khi quay lại, Bò phát hiện đàn bò đã vào vườn hoa và đang chén những cây hoa. Bò quyết định phải đưa lũ bò về chuồng ngay lập tức, hơn thế, còn phải dồn bò sao cho số hoa thiệt hại là ít nhất.

Con bò  $i$  đang ở địa điểm cách chuồng một khoảng  $T_i$  phút di chuyển, nó chén  $D_i$  cây hoa trong mỗi phút chờ tới lượt được đưa về chuồng. Ở mỗi thời điểm Bò chỉ có thể đưa một con bò về chuồng, với con bò  $i$  Bò sẽ mất  $2 \times T_i$  phút (đưa bò về chuồng và quay lại).

Lập chương trình xác định số cây hoa bị ăn nếu Bò dồn bò theo lịch tối ưu.

### Dữ liệu (flowers.inp)

- Dòng 1: ghi số nguyên  $N$  ( $2 \leq N \leq 100,000$ )
- Dòng 2 ...  $N + 1$ : mỗi dòng ghi hai số nguyên  $T_i$  ( $1 \leq T_i \leq 2,000,000$ ) và  $D_i$  ( $1 \leq D_i \leq 100$ ).

### Kết quả (flowers.out)

- Dòng 1: số nguyên là số cây hoa bị ăn ít nhất.

#### Ví dụ

flowers.inp	flowers.out
6 3 1 2 5 2 3 3 2 4 1 1 6	86

## Xếp thùng [stack]

Cho  $N$  thùng chứa đồ đánh số  $1 \dots N$ , thùng  $i$  có trọng lượng  $W_i$  và có thể chịu được tổng trọng lượng không quá  $C_i$  đặt trên. Hãy xác định số thùng nhiều nhất có thể xếp được thành một chồng.

### Dữ liệu (stack.inp)

- Dòng 1: số nguyên  $N$  ( $1 \leq N \leq 2,500$ )
- Dòng 2:  $N$  số nguyên  $W_1, W_2, \dots, W_N$  ( $1 \leq W_i \leq 100,000$ )
- Dòng 3:  $N$  số nguyên  $C_1, C_2, \dots, C_N$  ( $1 \leq C_i \leq 1,000,000,000$ )

### Kết quả (stack.out)

- Dòng 1: số nguyên là số thùng nhiều nhất có thể xếp thành một chồng

#### Ví dụ

stack.inp	stack.out
3 10 20 30 11 100 10	3
3 11 20 30 11 100 10	2

## Tàu điện (inftrain)

Tuyến tàu điện ZZZ có các ga được đánh số  $0, 1, 2, \dots$  theo thứ tự hành trình. Có tất cả  $N$  tàu điện hoạt động trên tuyến này. Tàu điện  $i$  có giới hạn số lượng hành khách  $c_i$ , xuất phát từ ga 0 chạy đến ga  $s_i$ . Các tàu điện đều xuất phát cùng một thời điểm và có thể đón, trả khách tại tất cả các ga mà nó đi qua.

Có  $M$  hành khách đang đợi tàu, hành khách  $i$  muốn lên tàu ở ga  $a_i$  và xuống ở ga  $b_i$ , mỗi hành khách chỉ đi một tàu.

Hãy xác định một phương án đón trả khách (hành khách nào lên tàu nào) sao cho đáp ứng được nhu cầu di chuyển của nhiều hành khách nhất.

### Dữ liệu

- Dòng 1: hai số nguyên  $N, M$  ( $1 \leq N, M \leq 10^5$ )
- Dòng 2 ...  $N + 1$ : dòng  $i + 1$  ghi hai số nguyên  $s_i, c_i$  ( $1 \leq s_i, c_i \leq 10^9 \forall i$ )
- Dòng  $N + 2$  ...  $N + M + 1$ : dòng  $i + N + 1$  ghi hai số nguyên  $a_i, b_i$  ( $1 \leq a_i \leq b_i \leq 10^9 \forall i$ ).

### Kết quả

- Dòng 1: số nguyên là số lượng hành khách phục vụ được.
- Dòng 2 ...  $N + 1$ : dòng  $i + 1$  ghi số nguyên là số hiệu tàu mà hành khách  $i$  lên, số này bằng 0 nếu hành khách này không được phục vụ.

### Ví dụ

inftrain.inp	inftrain.out
2 3	3
10 1	2
15 1	1
2 8	2
7 10	
8 13	

### Chấm điểm

- 20% điểm của bài dành cho các test có  $N = 1$ .

## Trình diễn xe hơi #1 (carshow1)

Hãng ZZZ tổ chức một buổi trình diễn  $N$  mẫu xe hơi mới của hãng.  $N$  xe sẽ cùng chạy trên một tuyến đường coi như trục tọa độ, xe  $i$  xuất phát ở thời điểm 0 từ vị trí  $x_i$  với vận tốc  $v_i$  đơn vị dài/giây chạy theo hướng dương của trục tọa độ, vị trí xuất phát của các xe là đôi một phân biệt.

Buổi trình diễn kéo dài trong  $T$  giây, vì đường chỉ có một làn nên khi có xe  $i$  đuổi kịp xe  $j$ , xe  $i$  cần giảm tốc độ bằng với xe  $j$  và chúng sẽ nối đuôi nhau chạy tiếp, coi kích thước các xe là không đáng kể. Như vậy ở mỗi thời điểm, trên đường sẽ có các nhóm xe chạy nối đuôi nhau mà ở thời điểm 0 có  $N$  nhóm (mỗi nhóm 1 xe).

Hãy xác định số nhóm xe chạy nối đuôi nhau ở thời điểm  $T$ .

### Dữ liệu

- Dòng 1: hai số nguyên  $N, T$  ( $1 \leq N \leq 10^5; 1 \leq T \leq 10^9$ )
- Dòng 2 ...  $N + 1$ : dòng  $i + 1$  ghi hai số nguyên  $x_i, v_i$  ( $0 < x_i; 0 < v_i; x_i, v_i \leq 10^9 \forall i$ ), các giá trị  $x_i$  là đôi một phân biệt, các xe được cho theo thứ tự  $x_i$  tăng.

## Kết quả

- Dòng 1: số nguyên kết quả.

## Ví dụ

carshow1.inp	carshow1.out
5 3 0 1 1 2 2 3 3 2 6 1	3

## Kéo xe [cardrag]

Bãi để xe ô tô bị thu giữ vì vi phạm luật giao thông có dạng lưới ô vuông kích thước  $M \times N$ , mỗi xe nằm gọn trên một ô vuông theo một trong bốn hướng N, S, E, W (bắc, nam, đông, tây).

Vì một số nguyên nhân đặc biệt, quản lý bãi đỗ cần giải phóng nhanh chóng càng nhiều xe càng tốt. Muốn vậy cần đẩy các xe ra khỏi bãi mà không gây va chạm, tức là cần đẩy các xe theo hướng đỗ ban đầu đi ra khỏi bãi sao cho trong quá trình đẩy xe không đi vào ô vẫn còn xe. Một xe nếu đã bị di chuyển thì phải được kéo ra khỏi bãi.

Hãy xác định số lượng xe nhiều nhất có thể đẩy ra khỏi bãi với một trình tự đẩy xe tối ưu.

## Dữ liệu

Gồm nhiều tests, mỗi test cho trên nhóm dòng theo định dạng:

- Dòng 1: hai số nguyên  $M, N$  ( $1 \leq M, N \leq 2000$ );
- Dòng 2 ...  $M + 1$ : mỗi dòng ghi xâu độ dài  $N$  chỉ gồm các kí tự trong tập  $\{ \cdot, N, S, E, W \}$  thể hiện ô trống hay ô có xe đỗ theo hướng tương ứng.

File dữ liệu kết thúc bằng dòng ghi hai số 0.

## Kết quả

Kết quả mỗi test ghi trên một dòng số nguyên là số lượng xe nhiều nhất có thể đẩy ra khỏi bãi.

## Ví dụ

cardrag.inp	cardrag.out
3 4 ·N·W WWSS EWEW 0 0	4

## Cân voi (scales)

Chán cảnh đi chăn trâu, Cuội bỏ lên miền núi chăn voi thuê cho nhà thống lí một bản nọ. Thống lí có một cô con gái rất xinh đẹp, lão "câu nhử" Cuội bằng cách giao hẹn nếu chăn đàn voi tốt thì sẽ gả con gái cho. Trời chiều lòng người, đàn voi Cuội chăn lớn nhanh như thổi. Giờ đây Cuội chỉ còn cân lũ voi nữa là ... xong.

Cuội được sử dụng một cân thăng bằng và  $N$  quả nặng đã biết khối lượng. Cuội cân voi bằng cách đưa voi lên một bàn cân và chất các quả nặng lên bàn cân bên kia cho đến khi thăng bằng.

(Cuội không đưa quả nặng lên bàn cân có voi được vì lũ voi này rất hiểu chiến, chúng sẽ dùng vòi hất ngay các quả nặng đi). Chiếc cân cũng có giới hạn, nó sẽ gãy nếu Cuội đặt lên một trong hai bàn cân tổng khối lượng vượt quá  $C$ .

Các quả nặng có một tính chất khá thú vị là nếu sắp chúng thành một hàng theo khối lượng tăng dần thì mỗi quả (từ thứ ba trở đi) sẽ không nhẹ hơn tổng khối lượng hai quả liền trước. Để tránh khả năng bị thống lí "đánh tháo" vì làm gãy cân, Cuội muốn biết khối lượng lớn nhất có thể cân được là bao nhiêu. Hãy lập trình tính toán giúp Cuội.

### Dữ liệu (scales.inp)

- Dòng 1: hai số nguyên  $N$   $C$  ( $1 \leq N \leq 1000$ ;  $1 \leq C < 2^{30}$ ).
- Dòng 2 ...  $N + 1$ : mỗi dòng một số nguyên trong phạm vi số nguyên 31 bit có dấu là khối lượng của một quả nặng, theo thứ tự không giảm.

### Kết quả (scales.out)

- Dòng 1: số nguyên chỉ khối lượng lớn nhất có thể cân được chính xác và an toàn.

### Ví dụ

scales.inp	scales.out
3 15 1 10 20	11